

Edge Intersection Graphs of L -Shaped Paths in Grids^{*}

Kathie Cameron¹ Steven Chaplick² Chính T. Hoàng³

¹ Department of Mathematics, Wilfrid Laurier University, Waterloo, On., Canada,
N2L 3C5, e-mail: kcameron@wlu.ca

² Department of Physics and Computer Science, Wilfrid Laurier University,
Waterloo, On., Canada, N2L 3C5, e-mail: chaplick@cs.toronto.edu

³ Department of Physics and Computer Science, Wilfrid Laurier University,
Waterloo, On., Canada, N2L 3C5, e-mail: choang@wlu.ca

Abstract. In this paper we continue the study of the edge intersection graphs of single bend paths on a rectangular grid (i.e., the edge intersection graphs where each vertex is represented by one of the following shapes: $\sqcup, \sqcap, \sqcup, \sqcap$). These graphs, called B_1 -EPG graphs, were first introduced by Golumbic et al (2009) [4]. We focus on the class \sqcup (the edge intersection graphs of \sqcup -shapes) and show that testing for membership in \sqcup is NP-complete. We then give a characterization and polytime recognition algorithm for special subclasses of $\text{Split} \cap \sqcup$. We also consider the natural subclasses of B_1 -EPG formed by the subsets of the four single bend shapes (i.e., $\{\sqcup\}, \{\sqcup, \sqcap\}, \{\sqcup, \sqcup\}, \{\sqcup, \sqcap, \sqcup\}$ – note: all other subsets are isomorphic to these up to 90 deg rotation). We observe the expected strict inclusions and incomparability (i.e., $\sqcup \subsetneq [\sqcup, \sqcap], [\sqcup, \sqcup] \subsetneq [\sqcup, \sqcap, \sqcup] \subsetneq B_1\text{-EPG}$ and \sqcup is incomparable with $[\sqcup, \sqcup]$).

1 Introduction

A graph G is called an *EPG graph* if G is the intersection graph of paths on a grid, where each vertex in G corresponds to a path on the grid and two vertices are adjacent in G iff the corresponding paths share an edge on the grid. EPG graphs were introduced by Golumbic et al [4]. The motivation for studying these graphs comes from circuit layout problems [2]. Golumbic and Jamison [5] proved that the recognition problem for the edge intersection graphs of paths in trees (EPT) is NP-complete even when restricted to chordal graphs. Golumbic et al [4] defined a B_k -EPG graph to be the edge intersection graph of paths on a grid where the paths are allowed to have at most k bends (turns). The B_0 -EPG graphs are exactly the well studied *interval graphs* (the intersection graphs of intervals on a line).

Heldt et al [6] proved that the recognition problem for B_1 -EPG is NP-complete. A graph is *chordal* if it does not contain a chordless cycle with at

^{*} Research support by Natural Sciences and Engineering Research Council of Canada.

least four vertices as an induced subgraph. A graph is a *split graph* if its vertices can be partitioned into a clique and a stable set. Asinowski and Ries [1] characterized special subclasses of chordal B_1 -EPG graphs.

Consider a B_1 -EPG graph G with a path representation on a grid. The paths can be of the following four shapes: $\sqsubset, \sqsupset, \sqcap, \sqcup$. In this paper, we study B_1 -EPG graphs whose paths on the grid belong to a proper subset the four shapes. If \mathcal{S} is a subset of $\{\sqsubset, \sqsupset, \sqcap, \sqcup\}$, then $[\mathcal{S}]$ denotes the class of graphs that can be represented by paths whose shapes belong to \mathcal{S} . We are interested in the class $[\sqsubset]$ of B_1 -EPG graphs whose paths are of the type \sqsubset . Our main results are:

- A proof of NP-completeness of recognition of $[\sqsubset]$.
- Characterizations of, and recognition algorithms for gem-free split $[\sqsubset]$ -graphs and bull-free split $[\sqsubset]$ -graphs, where *Split* denotes the class of split graphs.
- Establishment of expected separation between the classes: $[\sqsubset] \subsetneq [\sqsubset, \sqsupset], [\sqsubset, \sqcup] \subsetneq [\sqsubset, \sqsupset, \sqcup] \subsetneq B_1\text{-EPG}$ and the incomparability between $[\sqsubset, \sqsupset]$ and $[\sqsubset, \sqcup]$.

In section 2, we discuss background results and establish some properties of B_1 -EPG graphs. In section 3, we show that recognition of $[\sqsubset]$ is an NP-complete problem. In section 4, we give polytime recognition algorithms for the classes $[\sqsubset]$ of gem-free split graphs, and of bull-free split graphs. Finally, we conclude with some open questions in section 5.

2 Properties of B_1 -EPG graphs

Let \mathcal{P} be a collection of nontrivial simple paths on a grid \mathcal{G} . The *edge intersection graph* $EPG(\mathcal{P})$ has a vertex v for each path $P_v \in \mathcal{P}$ and two vertices are adjacent in $EPG(\mathcal{P})$ if the corresponding paths in \mathcal{P} share an edge of \mathcal{G} . For any grid edge e , the set of paths containing e is a clique in $EPG(\mathcal{P})$; such a clique is called an *edge-clique* [4]. A *claw* in a grid consists of three grid edges meeting at a grid point. The set of paths which contain two of the three edges of a claw is a clique; such a clique is called a *claw-clique* [4] (see figure 1).



Fig. 1. Left: An edge-clique. Right: A claw-clique.

Lemma 1 ([4]). *Consider a B_1 -EPG representation on a grid of a graph G . Every clique in G corresponds to either an edge-clique or a claw clique.*

An *asteroidal triple* (AT) is a set of three vertices such that for every pair, there is a path between them which avoids the neighbourhood of the other vertex.

Lemma 2 (AT Lemma [1], Theorem 9). *In a B_1 -EPG graph, no vertex can have an AT in its neighbourhood.*

Let C_4 denote the chordless cycle a, b, c, d, a on four vertices. Golumbic et al [4] proved that any B_1 -EPG representation of C_4 corresponds to what they call a “true pie”, a “false pie”, or a “frame”. True and false pies require paths other than \perp s. A *frame* is a rectangle in the grid \mathcal{G} such that each corner is the bend-point for one of P_a, P_b, P_c and P_d ; $P_a \cap P_b, P_b \cap P_c, P_c \cap P_d$, and $P_d \cap P_a$ each contain at least one edge; and $P_a \cap P_c$ and $P_b \cap P_d$ each do not contain an edge. Consider the C_4 and its four representations shown in figure 2. The first two representations are frames, the third is a false pie, and the fourth is a true pie. It follows that:

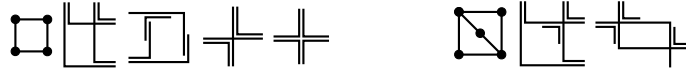


Fig. 2. Left: C_4 and its representations. Right: $K_{2,3}$ and its representations.

Lemma 3 (C_4 Lemma). *In the $[\perp]$ -representation of a C_4 every \perp has a neighbor on both its vertical segment and its horizontal segment.*

Observation 1 $K_{2,3}$ is in $[\perp, \top]$.

Proof. See figure 2 for a $[\perp, \top]$ -representation of $K_{2,3}$. □

Lemma 4 ($K_{2,3}$ Lemma). *In an $[\perp, \top]$ -representation of a $K_{2,3}$ every \perp (and \top) has a neighbour on both its vertical segment and on its horizontal segment.*

Proof. Consider $K_{2,3}$ to be the complete bipartite graph with bipartition $\{\{a, b\}, \{c, d, e\}\}$. Note that each of the following is a C_4 : a, c, b, d, a ; a, c, b, e, a ; and a, d, b, e, a . As noted above, any B_1 -EPG representation of C_4 corresponds to a “true pie”, a “false pie”, or a “frame”. True pies require paths all four types, but false pies and frames can be made from just \perp s and \top s.

If an $[\perp, \top]$ -representation of a C_4 corresponds to a frame, then every \perp (and \top) has a neighbour on both its vertical segment and on its horizontal segment. Consider an $[\perp, \top]$ -representation of a $K_{2,3}$.

Suppose first that both $\{a, c, b, d\}$ and $\{a, c, b, e\}$ correspond to frames. Then P_d and P_e must have the same bend-point, and this bend-point must be an intersection point of P_a and P_b . Since d and e are not adjacent, one of P_d and P_e is an \perp and the other is a \top . It follows that every \perp (and \top) has a neighbour on both its vertical segment and on its horizontal segment.

Now suppose that $\{a, c, b, d\}$ corresponds to a false pie. If P_a and P_b have the same bend-point, the bend-point must be an intersection point of P_c and P_d , and then there is nowhere to place P_e so that it intersects both P_a and P_b . So it must be that P_c and P_d have the same bend-point, which must be an intersection point of P_a and P_b , say point p . Then P_e must have bend-point at an intersection point of P_a and P_b , but since e is not adjacent to c or to d ,

this must be a different intersection point from p . So we have a configuration such as that in figure 2, and it follows that every \perp (and \top) has a neighbour on both its vertical segment and on its horizontal segment. (Note that $\{a, c, b, e\}$ corresponds to a frame.)

It is not possible for both $\{a, c, b, d\}$ and $\{a, c, b, e\}$ to both correspond to false pies. \square

Observation 2 $K_{2,3}$ is in $[\perp, \top]$ but not in $[\perp, \lceil]$.

Proof. Again, recall that a, c, b, d, a and a, c, b, e, a are C_4 s in $K_{2,3}$. True and false pies are not representable using just \perp s and \lceil s. So both of these must be represented as frames. As argued above, P_d and P_e must have the same bend-point. But since d and e are not adjacent, if P_d is an \perp , then P_e must be an \top and vice versa. It follows that $K_{2,3}$ is not in $[\perp, \lceil]$. \square



Fig. 3. Left: 3-sun and its representation. Right: 4-wheel and its representation

Observation 3 The 3-Sun is in $[\perp, \lceil]$ but not in $[\perp, \top]$.

Proof. See figure 3 for the 3-sun and an $[\perp, \lceil]$ -representation of the 3-sun. To see that the 3-sun does not have an $[\perp, \top]$ -representation, recall that in a B_1 -EPG graph, every clique is an edge-clique or a claw-clique. The vertices of the 3-sun can be partitioned into a clique with vertices a, b, c and a stable set with vertices d, e, f with edges da, dc, ea, eb, fb, fc . It is easy to see that if the clique $\{a, b, c\}$ is an edge-clique, then only two of d, e, f can be represented regardless of which types of 1-bend paths are used. So the clique $\{a, b, c\}$ is a claw-clique. But an \perp and \top can not be together in a claw-clique. \square

Observation 4 The 4-wheel is in $[\perp, \lceil, \top]$ but not in $[\perp, \lceil]$ or $[\perp, \top]$.

Proof. See figure 3 for the 4-wheel and an $[\perp, \lceil, \top]$ -representation of the 4-wheel. Lemma 3 in [1] shows that in a B_1 -representation of W_4 , the C_4 corresponds to a true pie or false pie. Since the true pie requires four shapes, we may assume the C_4 of the W_4 is represented by a false pie. So, W_4 is not an $[\perp, \lceil]$ -graph. Consider the vertex u of W_4 that is adjacent to all vertices of the C_4 . If P_u is of type \perp or \top , then P_u can not share an edge grid with all four paths of the C_4 . So, the W_4 is not an $[\perp, \top]$ -graph. \square

3 NP-Hardness: Recognition of \llcorner

It is well-known that interval graphs (i.e., B_0 -EPG graphs) can be recognized in polynomial time [3]. The complexity of the recognition problem for B_k -EPG ($k > 0$) was given as an open problem in the paper introducing EPG graphs [4]. The recognition problem for B_1 -EPG has been shown to be NP-complete in a recent paper [6]. In this section we consider the complexity of recognizing the simplest natural subclass of B_1 -EPG which is a superclass of B_0 -EPG; namely, \llcorner . Specifically, we show that it is NP-complete to decide membership in \llcorner .

Theorem 5. *Deciding membership in \llcorner is NP-complete.*

Proof. A given \llcorner model is easily verified, so \llcorner recognition is in NP. For NP-hardness we demonstrate a reduction from the usual 3-SAT problem (defined below). Our reduction is inspired by the NP-completeness proof for B_1 -EPG [6].

The essential ingredients of our construction are described in the following observations. In an \llcorner -representation H of a graph G with vertices u, v , we say that v is an *internal neighbor* of u in H when: v is adjacent to u , P_u 's bend-point is not contained in P_v and w.l.o.g. P_u 's horizontal contains P_v 's horizontal (see figure 4(i)). We also say that v is an *external neighbor* of u when v is adjacent to u but v is not an internal neighbor of u . Notice that, in any \llcorner -representation of a graph a vertex can have at most four stable external neighbors (as depicted in figure 4(ii)). Additionally, if a vertex v is an internal neighbor of a vertex u , then v can have at most two stable external neighbors which are not adjacent to u (see figure 4(iii)). Finally, we say that a vertex u is *adjacent to a C_4* when u is adjacent to exactly one vertex in an induced C_4 (see figure 4(iv)). Now consider a graph G with a vertex u which is adjacent to a C_4 and let v be u 's neighbor in this C_4 . Recall that, in any \llcorner -representation of an induced C_4 , every \llcorner -path has a neighbor with an edge intersection on its vertical and a neighbor with an edge intersection on its horizontal (by Lemma 3). Thus, in any \llcorner -representation of G , v is necessarily an external neighbor of u . With these observations in mind we can now describe the structure of our graph G_Φ . A 3-SAT formula Φ is a

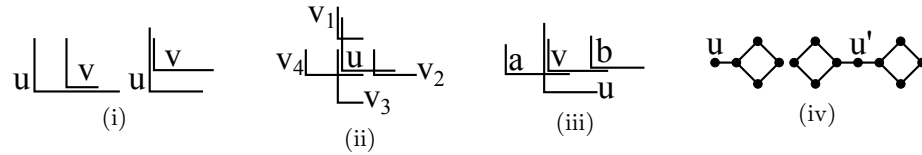


Fig. 4. (i): v is an internal neighbor of u (left: *internal horizontal neighbor*; right: *internal vertical neighbor*). (ii): u with stable external neighbors v_1, v_2, v_3, v_4 . (iii): v is an internal neighbor of u , and v has two stable external neighbors a, b which are not adjacent to u . (iv): u adjacent to one C_4 and u' adjacent to two adjacent C_4 s.

boolean formula over variables x_1, \dots, x_k where Φ is a conjunction of t clauses (D_1, D_2, \dots, D_t) , each clause D_i ($1 \leq i \leq t$) is a disjunction of three literals

$(\ell_{i1}, \ell_{i2}, \ell_{i3})$, and each literal ℓ_{iq} ($1 \leq q \leq 3$) is either the negation or non-negation of some variable x_j ($1 \leq j \leq k$). Given a 3-SAT formula Φ , it is well known that it is NP-complete to decide whether there exists an assignment to the variables of Φ that satisfies Φ [7].

Given a 3-SAT formula Φ we will construct a graph G_Φ such that G_Φ is an $[\perp]$ -graph iff Φ can be satisfied. G_Φ consists of an induced subgraph G_{D_i} for each clause D_i of Φ and a variable gadget to identify the clauses with their corresponding literals. The general form of these gadgets is given in figure 5. We begin by

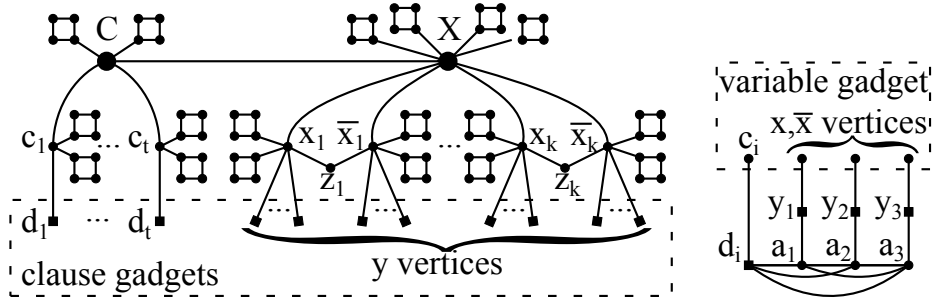


Fig. 5. The general form of G_Φ . Variable gadget (left); clause gadget (right). Note: a literal (i.e., x_j or $\overline{x_j}$) is adjacent to y_q (in a clause gadget G_{D_i}) when that literal is in D_i . Also, each y_q is adjacent to exactly one literal.

describing the structure of the $[\perp]$ -representation of the variable gadget. Notice that the vertex X is adjacent to four C_4 s. Thus, as we have observed, X will have four external neighbors in any $[\perp]$ -representation of G_Φ . Furthermore, since the neighborhood of X is a stable set, the vertices $C, x_1, x_2, \dots, x_k, \overline{x_1}, \overline{x_2}, \dots, \overline{x_k}$ are all internal neighbors of X . Finally, suppose that x_j is an internal horizontal neighbor of X . Since $G_\Phi[\{X, x_j, \overline{x_j}, z_j\}]$ is a C_4 , $\overline{x_j}$ is necessarily an internal vertical neighbor of X . Similarly, if x_j were to be an internal vertical neighbor of X , $\overline{x_j}$ would necessarily be an internal horizontal neighbor of X ⁽⁴⁾. From these observations we depict the general structure of an $[\perp]$ -representation of the subgraph of G_Φ induced by $\{X, x_1, \dots, x_k, \overline{x_1}, \dots, \overline{x_k}, z_1, \dots, z_k\}$ and the C_4 s adjacent to these vertices in figure 6.

Now, w.l.o.g., suppose that C is an internal horizontal neighbor of X . Notice that C is adjacent to two C_4 s, is an internal horizontal neighbor of X , and the neighborhoods of X and C are disjoint. Thus, since the neighborhood of C is a stable set, the vertices c_1, \dots, c_t are internal vertical neighbors of C . Similarly, for each $1 \leq i \leq t$, d_i is an internal horizontal neighbor of c_i since each c_i is an internal vertical neighbor of C and each c_i is adjacent to two C_4 s. These observations provide the general structure of an $[\perp]$ -representation of the subgraph of G_Φ

⁴ We will later use the location (i.e., as an internal horizontal or internal vertical neighbor of X) as a variable's truth value.

induced by $\{X, C, c_1, \dots, c_t, d_1, \dots, d_t\}$ and the C_4 s adjacent to these vertices (as seen in figure 6). With the restricted structure of the variable gadget in mind, we

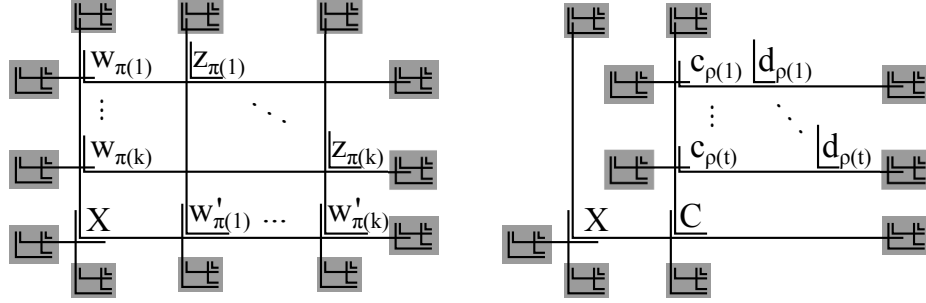


Fig. 6. Left: The possible $\lfloor \cdot \rfloor$ -representations of G_Φ induced by $\{X, x_1, \dots, x_k, \overline{x_1}, \dots, \overline{x_k}, z_1, \dots, z_k\}$ and the C_4 s adjacent to these vertices (note: $w_i \in \{x_i, \overline{x_i}\}$ and $\{w_i, w'_i\} = \{x_i, \overline{x_i}\}$, and π is a permutation on $\{1, \dots, k\}$). Right: The possible $\lfloor \cdot \rfloor$ -representations of G_Φ induced by $\{X, C, c_1, \dots, c_t, d_1, \dots, d_t\}$ and the C_4 s adjacent to these vertices (note: ρ is a permutation on $\{1, \dots, t\}$).

now turn our attention to the clause gadget of a clause $D_i = (\ell_1, \ell_2, \ell_3)$. Notice that $\{d_i, a_1, a_2, a_3\}$ is a clique (i.e., $\{P_{d_i}, P_{a_1}, P_{a_2}, P_{a_3}\}$ form an edge-clique in any $\lfloor \cdot \rfloor$ -representation of G_Φ). Furthermore, $\{P_{d_i}, P_{a_1}, P_{a_2}, P_{a_3}\}$ form a vertical edge-clique since d_i is an internal horizontal neighbor of c_i and c_i is not adjacent to any of a_1, a_2 , or a_3 . Notice that, in a vertical edge-clique, there can be at most two $\lfloor \cdot \rfloor$ -paths which contain vertical grid edges that are not contained in the other $\lfloor \cdot \rfloor$ -paths of the edge-clique (i.e., the “top-most” and bottom-most $\lfloor \cdot \rfloor$ -paths – e.g., the clique $\{d_i, a_1, a_2, a_3\}$ in figure 7). Thus, w.l.o.g., we suppose that P_{a_1} and P_{a_3} have this property. In particular, this means that P_{a_2} and P_{y_2} necessarily intersect via a horizontal grid edge and no vertical grid edge (since every grid edge contained in P_{a_2} is also contained in one of P_{a_1} or P_{a_3} and y_2 is adjacent to neither a_1 nor a_3). Additionally, observe that, when ℓ_q ⁵ is an internal vertical neighbor of X , y_q is necessarily an internal horizontal neighbor of ℓ_q since ℓ_q is adjacent to two C_4 s. Similarly, when ℓ_q is an internal horizontal neighbor of X , y_q is an internal vertical neighbor of ℓ_q . However, y_2 cannot be an internal horizontal neighbor of ℓ_2 since ℓ_2 is not adjacent to a_2 and P_{y_2} and P_{a_2} have a horizontal grid edge in common. Thus, it is not possible for all three literals to be internal vertical neighbors of X . On the other hand, when at most two literals are internal vertical neighbors of X , we can always construct the $\lfloor \cdot \rfloor$ -representation of the clause gadget. In particular, this can be done using one of the three templates depicted in figure 7. Note, to form an $\lfloor \cdot \rfloor$ -representation of G_Φ , the placement of the $\lfloor \cdot \rfloor$ -representations of the clause gadgets from figure 7 can be described as follows:

⁵ Remember, ℓ_q is some x_j or $\overline{x_j}$ ($1 \leq j \leq k$).

- For type (i) and (ii) (i.e., at most one literal is an internal vertical neighbor of X), we place the $[\perp]$ -representation of the clause gadget “below” $P_{w_{\pi(k)}}$ and to the “left” of $P_{w'_{\pi(1)}}$ (with respect to the depiction in figure 6).
- For type (iii) (i.e., two literals ℓ_1 and ℓ_3 which are internal horizontal neighbors of X), we need to place the $[\perp]$ -representation of the clause gadget “between” P_{ℓ_1} and P_{ℓ_3} and to the “left” of $P_{w'_{\pi(1)}}$ (with respect to the depiction in figure 6).

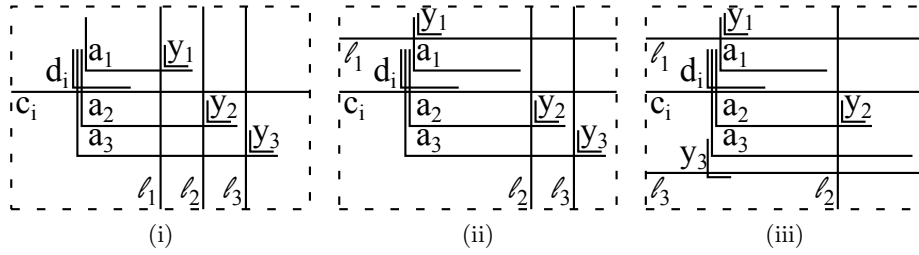


Fig. 7. $[\perp]$ -representations of the clause gadget for a clause (ℓ_1, ℓ_2, ℓ_3) inside an $[\perp]$ -representation of G_Φ . (i) $\ell_1 = \ell_2 = \ell_3 = \text{true}$; (ii) $\ell_1 = \text{false}$ and $\ell_2 = \ell_3 = \text{true}$; (iii) $\ell_1 = \ell_3 = \text{false}$ and $\ell_2 = \text{true}$.

We can now see that a literal being an internal vertical neighbor of X corresponds to when that literal is *false* (since at most two literals can be internal vertical neighbors of X) and a literal being an internal horizontal neighbor of X corresponds to when that literal is *true*. Thus, since x_j and $\overline{x_j}$ cannot both be internal vertical (or horizontal) neighbors of X , the $[\perp]$ -representations of the graph G_Φ correspond to satisfying assignments of Φ . \square

An interesting observation regarding this proof is that it can be easily adapted to show the NP-completeness of recognition for $[\perp, \lceil]$ and $[\perp, \sqcap]$. For $[\perp, \lceil]$, the same graph G_Φ can be used, but one has to be careful about the structure of the $[\perp, \lceil]$ -representation of the clause gadget (since it need not be an edge clique). For $[\perp, \sqcap]$, we alter G_Φ slightly. First, we replace each C_4 adjacent to a vertex with a $K_{2,3}$ adjacent to the same vertex (by Lemma 4, this forces the vertex to have an external neighbor for each $K_{2,3}$, just as we had with the C_4 s in $[\perp]$ -representations). Second, for each $1 \leq i \leq k$, we add a vertex z'_i which is adjacent to x_i and $\overline{x_i}$ (thus, turning the C_4 s induced by $\{X, x_i, \overline{x_i}, z_i\}$ into $K_{2,3}$ s and preventing x_i and $\overline{x_i}$ from both being internal vertical (horizontal) neighbors of X). These two changes to G_Φ allow the proof to proceed as before. We conjecture that similar adaptations can be performed to demonstrate the NP-completeness of $[\perp, \lceil, \sqcap]$ and possibly even B_k -EPG for $k > 1$.

4 Characterization and Recognition of $Split \cap [\perp]$

Recall that recognizing chordal EPT graphs is NP-complete. We have just shown that recognizing $[\perp]$ -graphs is NP-complete. Thus, it is of interest to characterize the class $Chordal \cap [\perp]$. A first step in this direction would be to study $Split \cap [\perp]$, that is, the class of split $[\perp]$ -graphs. We split this discussion into three parts. In the first part, we establish some properties of split $[\perp]$ -graphs. In the latter two parts, we characterize two special subclasses of split $[\perp]$ -graphs.

4.1 Properties of $Split \cap [\perp]$

In this section, we will establish some properties of the class $Split \cap [\perp]$. We will pose a conjecture on the characterization of this class. First, we need to introduce a few definitions.

A vertex x *dominates* a vertex y if $N(y) \subseteq N(x) \cup \{x\}$, where $N(a)$ denotes the set of vertices adjacent to a vertex a . Vertex x is *comparable* to vertex y if x dominates y , or vice versa. The domination relation is a partial order. A vertex is *maximal* if it is not dominated by another vertex. If vertex x belongs to a set X of vertices of G , we will say that x is an X -vertex. We let $G - X$ denote the subgraph of G induced by the vertices belonging to $V(G)$ but not to X . We let $N(X)$ denote the set of vertices outside X that have some neighbors in X . Two vertices a, b are *twins* if $N(a) \cup \{a\} = N(b) \cup \{b\}$ or $N(a) = N(b)$. A *split partition* (C, S) of a graph G is a partition of its vertices into a clique C and a stable set S . We will enumerate the vertices of S as $\{s_1, \dots, s_k\}$. In this section, we consider split graphs that admit $[\perp]$ -representations.

Let G be an $[\perp]$ -graph with a split partition (C, S) . It follows from Theorem 1 that C corresponds to an edge-clique. Consider an $[\perp]$ -representation of G on the grid. We may assume without loss of generality that the edge of the grid that belongs to all of C is vertical. The horizontal parts of paths of C are called *branches*. The vertical part of C below the first (top) branch is called the *trunk*. The vertical part of C above the first branch is called the *crown* (see figure 8).

Observation 6 *The vertices of S whose \perp -paths lie on the same branch (or, the crown) are pairwise comparable. An S -vertex whose path lies on the trunk dominates all S -vertices whose paths lie below it in the representation. \square*

See figure 8 for an illustration of Observation 6. The *gem* is the graph with vertices a, b, c, d, e , edges $ab, bc, cd, ea, eb, ec, ed$. The *bull* is the graph with vertices a, b, c, d, e , edges ab, bc, cd, eb, ec .

Observation 7 *Let G be a split graph with a split partition (C, S) . If G admits an $[\perp]$ -representation and contains a gem, then exactly one of the gem's S -vertices has its \perp -path occurring on the crown of the representation.*

Proof. Let the vertices of the gem be c_1, c_2, c_3, s_1, s_2 with $c_1, c_2, c_3 \in C$, $s_1, s_2 \in S$ and $s_1c_1, s_1c_2, s_2c_2, s_2c_3 \in E(G)$. Assume that both P_{s_i} are not on the crown.



Fig. 8. A Split $\cap[L]$ graph (left) and an $[L]$ -representation of it (right).

Since s_1 and s_2 are incomparable, by Observation 6, we may assume P_{s_2} lies on a branch. Since s_1 is adjacent to c_2 , P_{s_1} must lie on the vertical segment of P_{c_2} and be above P_{c_2} in the representation. By our assumption, P_{s_1} must be on the trunk. By Observation 6, s_1 dominates s_2 , a contradiction. Thus, we may assume P_{s_1} is on the crown. Since s_1 is incomparable with s_2 , P_{s_2} cannot be on the crown. \square

Definition 1. An S -bull is a bull such that the three vertices of degrees less than three in the bull are in S .

In figure 8, $\{b, c, 1, 2, 3\}$ is an S -bull but $\{a, b, c, 5, 6\}$ is not an S -bull even though it is a bull.

Observation 8 Let G be a split graph with a split partition (C, S) . If G admits an $[L]$ -representation and contains an S -bull, then some S -vertices of this bull have their paths occurring on either the crown or trunk of the representation. \square

Observation 9 Let G be a split graph with a split partition (C, S) . Suppose there is a vertex v in G with $N(v) = C - \{v\}$. Then G is an $[L]$ -graph iff $G - v$ is.

Proof. Suppose $G - v$ has an $[L]$ -representation. First assume $v \in S$. On the trunk, there is a vertical segment where all of C meets. We can place P_v there to get a representation of G . Now, we may assume $v \in C$. Thus, v has no neighbor in S . We can place P_v on the grid edge at the base of the crown—and if necessary move the other paths of S on the crown up—to obtain a representation for G . \square

Observation 10 Let G be a split graph with a split partition (C, S) . Suppose G contains twins a, b . Then G is an $[L]$ -graph iff $G - a$ is.

Proof. Suppose a is adjacent to b . Suppose further that a is in S . Then b is in C and it follows that a is adjacent to all vertices of C . So, we are done by Observation 9. Thus, both a and b are in C . Consider an $[L]$ -representation of $G - a$. By making P_a an exact copy of P_b , we obtain a representation for G .

So, we may assume a is not adjacent to b . Suppose both a and b are in S . Consider an $[L]$ -representation of $G - a$. Then P_b occurs on a branch (trunk, crown). By placing P_a next to P_b on this branch (trunk, crown), we obtain a representation for G (see figure 8 for an illustration.) Now, we may assume a is in C and b is in S . It follows that a has no neighbor in S . But then we are done by Observation 9. \square

Observation 11 Let G be a split graph with a split partition (C, S) . Suppose there is a subset D of C such that the vertices of $X = N(D) \cap S$ are pairwise comparable and $N(S) \subseteq D$. Then G is an $[\perp]$ -graph iff $G - (D \cup X)$ is.

Proof. Suppose there is an $[\perp]$ -representation of $G - (D \cup X)$. Vertices of D will be represented by \perp -paths with the same bend-point. Recall that the \perp -paths of C lie on the trunk. We will add the vertical segments of the \perp -paths D to the trunk. We can move the \perp -paths of the S -vertices at the crown up so they do not intersect with the vertical parts of the paths of D whose bend-point is placed just below the first branch. We can place the vertices (paths) of X on this new branch. \square

Observation 12 Let G be a split graph with a split partition (C, S) . Suppose some vertex $c \in C$ is such that all of its neighbors in S have degree one. Then G is an $[\perp]$ -graph iff $G - c$ is. \square

Observation 13 Let G be a gem-free graph with a split partition (C, S) . Then any two vertices of S with a common neighbor in C are comparable. \square

Observation 14 Let G be a gem-free graph with a split partition (C, S) . Let s be a maximal vertex in S and s' be a vertex in S with a common neighbor with s . Then s dominates s' . \square

Consider the nine graphs shown in figure 9. We believe that they are the only minimal forbidden obstructions for a split graph to be an $[\perp]$ -graph.

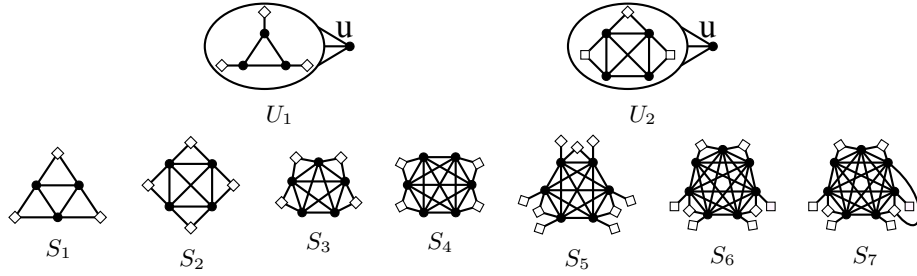


Fig. 9. In U_1 and U_2 , the vertex u is adjacent to all remaining vertices.

Lemma 5. None of the nine graphs shown in figure 9 is an $[\perp]$ -graph.

Proof. By Lemma 1, the graphs U_1 and U_2 do not admit $[\perp]$ -representations. Consider the graph S_1 with the split partition (C, S) where $C = \{c_1, c_2, c_3\}$, $S = \{s_1, s_2, s_3\}$, and $v_i c_i, v_i c_{i+1} \in E(G)$ with the subscripts taken modulo 3. Now, consider the gem $\{s_1, s_2, c_1, c_2, c_3\}$. By Observation 7, we may assume P_{s_1} is on the crown and P_{s_2} is not. The gem $\{s_1, s_3, c_1, c_2, c_3\}$ implies that P_{s_3} is

not on the crown because P_{s_1} already is. But the two S -vertices of the gem $\{s_2, s_3, c_1, c_2, c_3\}$ have no paths on the crown, a contradiction to Observation 7. So, S_1 is not an $[\perp]$ -graph. Similar arguments show S_2 , S_3 , and S_4 are not $[\perp]$ -graphs. Consider the graph S_5 . Suppose S_5 admits an $[\perp]$ -representation. Let B_1, B_2, B_3 be the three S -bulls of S_5 . By Observation 8, each B_i contains an S -vertex s_i such that P_{s_i} lies on the trunk or crown. Without loss of generality, we may assume the trunk contains s_1 and s_2 . The fact that s_1 is incomparable with s_2 contradicts Observation 6. Similar arguments show that S_6 and S_7 are not $[\perp]$ -graphs. Finally, it is a routine but tedious matter to show that all proper induced subgraphs of the graphs in figure 9 are $[\perp]$ -graphs. \square

We believe the nine graphs in figure 9 are the only minimal forbidden subgraphs for $\text{Split} \cap [\perp]$. We pose that as a conjecture.

Conjecture 1. A split graph is an $[\perp]$ -graph iff it does not contain any of the nine graphs in figure 9 as an induced subgraph.

Theorems 17 and 18 (proved in the next sections) can be seen as first steps in this direction.

The k -sun ($k \geq 3$) is the graph obtained by taking a cycle on $2k$ vertices and joining odd-indexed vertices by edges. So, a 3-sun is the graph S_1 , a 4-sun is the graph S_2 , and S_3 occurs in any k -sun with $k \geq 5$. A graph is *strongly chordal* if it is chordal and contains no k -sun. The following lemma follows from Lemma 5.

Observation 15 $\text{Chordal} \cap [\perp] = \text{Strongly Chordal} \cap [\perp]$. \square

4.2 Split graphs without S-bulls

In this section, we give a characterization by forbidden induced subgraphs of split $[\perp]$ -graphs without S -bulls. This provides a polytime algorithm for recognizing split $[\perp]$ -graphs without S -bulls.

Observation 16 Let x_1, x_2 be two incomparable vertices in S . If G does not contain an S -bull, then no vertex $s \in S$ is adjacent to some vertex x of $N(x_1) - N(x_2)$ and some vertex y of $N(x_2) - N(x_1)$.

Proof. If such vertex s exists, then $\{s, x, y, x_1, x_2\}$ induces a S -bull. \square

Theorem 17. Let G be a graph with a split partition (C, S) and with no S -bull. Then G admits an $[\perp]$ -representation iff G does not contain U_1 or S_4 , as an induced subgraph.

Proof. By induction on the number of vertices. We only need to prove the “if” part. Let G be a graph with a split partition (C, S) and with no S -bull, U_1 , or S_4 . Suppose no two incomparable S -vertices have a common neighbor (ie., G is gem-free). Then, by Observation 16, there is a partition of S into sets X_1, X_2, \dots, X_k ($k \geq 2$) such that the vertices of X_i are pairwise comparable and $N(u) \cap N(v) = \emptyset$ for any $u \in X_i, v \in X_j, i \neq j$. By the induction hypothesis, the

graph $G' = G - (X_1 \cup N(X_1) \cap S)$ admits an \lfloor -representation. By Observation 11, G admits an \lfloor -representation. (Note: this implies Corollary 1 below.)

So, there are two incomparable S -vertices with a common neighbor. Let $s_1, s_2 \in S$ be two incomparable vertices with a common neighbor such that $d(s_1) + d(s_2)$ is largest, where $d(x)$ denotes the degree of vertex x . The following two facts are easy to establish.

Let s_3 be an S -vertex with a neighbor in $N(s_1) \cup N(s_2)$. Then s_3 is comparable to s_1 or to s_2 . (1)

Suppose s_3 is incomparable to both s_1 and s_2 . Vertex s_3 has no neighbors in $N(s_1) \cap N(s_2)$, for otherwise G contains an universal AT, and thus an S -bull or U_1 . Without loss of generality, we may assume s_3 has a neighbor x in $N(s_1) - N(s_2)$. Now, there is a S -bull with vertices s_1, s_2, s_3, x , and some $y \in N(s_1) \cap N(s_2)$. We have established (1).

Define $C_0 = N(s_1) \cup N(s_2)$.

For any vertex $s_3 \in S$ with a neighbor in C_0 , either s_1 or s_2 dominates s_3 . (2)

Consider a vertex $s_3 \in S$ with a neighbor in C_0 . Suppose s_3 has a neighbor $y \notin C_0$. By (1), we may assume s_3 is comparable to s_2 . The existence of y implies s_3 dominates s_2 . It follows that s_3 is comparable to s_1 , for otherwise, $d(s_3) + d(s_1) > d(s_2) + d(s_1)$, contradicting our choice of s_1 and s_2 . Thus, s_3 dominates s_1 . By Observation 16, with $x_1 = s_1, x_2 = s_2, s_3 = s$, G contains an S -bull, a contradiction. So, we have $N(s_3) \subseteq C_0$. By Observation 16, s_3 has no neighbor in $N(s_1) - N(s_2)$ or in $N(s_2) - N(s_1)$. Thus, (2) is established.

We show the placements of the paths of $N(C_0) \cap S$ on the crown and first branch. We place the paths of $S - N(C_0)$ on branches below that first branch. By (2), the vertices of $N(C_0) \cap S$ can be partitioned into two sets D_1 and D_2 such that s_i is in D_i and dominates every vertex in $D_i - s_i$. Now, we claim that

The vertices in each D_i are pairwise comparable. (3)

If some two vertices $x_1, x_2 \in D_i$ are incomparable, then by Observation 16, G contains an S -bull. So, (3) holds. It follows that the vertices of C_0 are pairwise comparable in the subgraph of G induced by $C_0 \cup D_1$ (and, $C_0 \cup D_2$). Vertices of C_0 will be represented by \lfloor -paths with the same bend-point. Place the paths representing D_1 at the crown with P_x being above P_y if x is dominated by y . Place the paths representing D_2 on the first branch with P_x to the right of P_y if x is dominated by y . For any two vertices a, b of D_1 (resp., D_2), if a dominates b in D_1 (resp., D_2), then every \lfloor -path of a C -vertex must pass through an edge of P_a to reach P_b . This completes the description of the representation of $C_0 \cup (N(C_0) \cap S)$.

Define $C' = C - C_0$. By (2), there is no vertex in S with a neighbor in C' and one in C_0 . The set $C' \cup (N(C') \cap S)$ contains no gem, for otherwise, G contains S_4 . It follows from Observations 13 and 14 that the set C' can be partitioned into sets C_1, C_2, \dots, C_k ($k \geq 1$) such that, for each i , the vertices

in $N(C_i) \cap S$ are pairwise comparable, and no S -vertex has a neighbor in C_i and one in C_j , for $i \neq j$ (in particular, for each C_i , there is a maximal S -vertex s with $N(s) \cap C = C_i$). Define $X = N(C_1) \cap S$. By the induction hypothesis, $G - (C_1 \cup X)$ is an \sqsubseteq -graph. By Observation 11, G is an \sqsubseteq -graph. \square

We note a polytime algorithm to construct an \sqsubseteq -representation for the input graph can be extracted from the proofs above. The algorithm is certifying in the sense that it produces either an \sqsubseteq -representation, or an obstruction. The proof of the theorem also implies the following corollary:

Corollary 1. *All S -bull free, gem-free split graphs are \sqsubseteq -graphs.* \square

4.3 Split graphs without gems

In this section, we give a characterization by forbidden induced subgraphs of split \sqsubseteq -graphs without gems. This provides a polytime algorithm for recognizing split \sqsubseteq -graphs without gems. First, we need to introduce a definition. Two vertex-disjoint S -bulls are *incomparable* if every S -vertex in one bull is incomparable with every S -vertex in the other bull.

Lemma 6. *Let G be a gem-free graph with a split partition (C, S) . Suppose G does not contain two incomparable S -bulls. Then, there is an \sqsubseteq -representation of G with no S -vertices having their \sqsubseteq -paths lying on the trunk.*

Proof. By induction on the number of vertices. We may assume every vertex of C has a neighbor in S , for otherwise we are easily done by the induction hypothesis. Let s be a maximal vertex in S .

Suppose $N(s) = C$. By the induction hypothesis, there is an \sqsubseteq -representation of $G - s$ with no paths of $S - s$ on the trunk. We can place P_s at the lowest part of the crown to obtain the desired \sqsubseteq -representation for G .

So, we have $C - N(s) \neq \emptyset$. By Observation 14, C contains at least two proper subsets A, B such that no S -vertex has some neighbors in A and some in B . It follows C contains a proper subset C' such that $C' \cup S'$ contains no S -bull, where $S' = N(C') \cap S$. We may choose C' such that the edges between C' and S' form a connected subgraph of G . It follows that the vertices of S' are pairwise comparable. By the induction hypothesis, $G - (C' \cup S')$ admits a \sqsubseteq -representation with no paths representing $S - S'$ on the trunk. We now add the paths of C' to the trunk making them share the same bend-point which is placed just below the first branch. By moving the paths of the S -vertices on the crown up, we can preserve the required adjacency between them and the new paths. We can place the paths of S' on the new branch, introducing no new paths on the trunk. \square

Theorem 18. *Let G be a gem-free graph with a split partition (C, S) . Then G admits an \sqsubseteq -representation iff G does not contain S_5 as induced subgraphs.*

Proof. By induction on the number of vertices. We only need to prove the “if” part. Let G be a gem-free graph with a split partition (C, S) and not containing S_6 . By Observation 9, we may assume every vertex in C has a neighbor in S . Let x be a maximal vertex in S . By Observation 9, we may assume $C - N(x) \neq \emptyset$. It follows from Observation 14 that S contains vertices x_1, x_2, \dots, x_k ($k \geq 2$) such that no vertex $s \in S$ has a neighbor in $N(x_i)$ and one in $N(x_j)$, $i \neq j$. Define $C_i = N(x_i)$, $S_i = N(C_i) \cap S$, for $i = 1, 2, \dots, k$. If for some C_i , the vertices of S_i are pairwise comparable, then we are done by the induction hypothesis and Observation 11. Therefore, for all i , S_i must contain two incomparable vertices. That is, each set $C_i \cup S_i$ must contain an S -bull. It follows that $k = 2$, and we may assume furthermore that $G_1 = G[C_1 \cup S_1]$ does not contain two incomparable S -bulls, for otherwise, G contains S_5 . By Lemma 6, there is a $[\perp]$ -representation of G_1 with no vertices of S_1 on the trunk. By the induction hypothesis, the graph $G_2 = G - (C_1 \cup S_1)$ has an $[\perp]$ -representation. We place the branches of G_2 under those of G_1 and extend the vertical segments of the paths of $C - C_1$ to the crown of G_1 . The adjacency of G is preserved because G_1 has no S -vertices on the trunk in the $[\perp]$ -representation. \square

5 Concluding Remarks and Open Problems

In this paper, we considered the edge intersection graphs of $[\perp]$ -shaped paths on a grid. We showed that recognizing such graphs is NP-complete. We considered the open problem of characterizing chordal $[\perp]$ -graphs. As first steps in solving this problem, we found characterizations of split gem-free $[\perp]$ -graphs and split $[\perp]$ -graphs without S -bulls (a class more general than split bull-free $[\perp]$ -graphs). Our characterizations imply polytime algorithms for recognizing these two classes of graphs. We posed a conjecture on the characterization of split $[\perp]$ -graphs. This conjecture would imply a polytime recognition theorem for split $[\perp]$ -graphs. The following open problems related to our works arise: (1) Extending the observations in section 4 to other subclasses of B_1 -EPG graphs; (2) Find a polytime algorithm for $Chordal \cap [\perp]$; (3) Establish NP-completeness for recognition of B_k -EPG graphs for every k at least 2.

References

1. ASINOWSKI, A., AND RIES, B. Some properties of edge intersection graphs of single-bend paths on a grid. *Discrete Mathematics* 312, 2 (2012), 427–440.
2. BANDY, M., AND SARRAFZADEH, M. Stretching a knock-knee layout for multilayer wiring. *IEEE Transactions on Computers* 39 (1990), 148151.
3. BOOTH, K. S., AND LUEKER, G. S. Testing for the consecutive ones property, interval graphs, and graph planarity using pq-tree algorithms. *Journal of Computer and System Sciences* 13, 3 (1976), 335–379.
4. GOLUMBIC, M., LIPSHTEYN, M., AND STERN, M. Edge intersection graphs of single bend paths on a grid. *Networks* 54 (2009), 130–138.
5. GOLUMBIC, M. C., AND JAMISON, R. E. The edge intersection graphs of paths in a tree. *J. Comb. Theory, Ser. B* 38, 1 (1985), 8–22.

6. HELDT, D., KNAUER, K., AND UECKERDT, T. Edge-intersection graphs of grid paths: the bend-number. unpublished manuscript, <http://arxiv.org/abs/1009.2861>.
7. KARP, R. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, R. Miller and J. Thatcher, Eds. Plenum Press, 1972, pp. 85–103.